

This Page Is Inserted by IFW Operations
and is not a part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

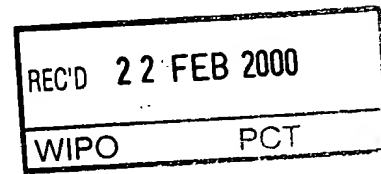
IMAGES ARE BEST AVAILABLE COPY.

**As rescanning documents *will not* correct images,
please do not report the images to the
Image Problem Mailbox.**

THIS PAGE BLANK (USPTO)

BUNDESREPUBLIK DEUTSCHLAND *ETU*

**PRIORITY
DOCUMENT**
SUBMITTED OR TRANSMITTED IN
COMPLIANCE WITH RULE 17.1(a) OR (b)

*EP00/386***Bescheinigung**

Die Deutsche Telekom AG in Bonn/Deutschland hat eine Patentanmeldung unter der Bezeichnung

"Verfahren zur grafischen Darstellung und/oder Bearbeitung von
Werten von Datentypen"

am 11. Februar 1999 beim Deutschen Patent- und Markenamt eingereicht.

Die angehefteten Stücke sind eine richtige und genaue Wiedergabe der ursprünglichen Unterlagen dieser Patentanmeldung.

Die Anmeldung hat im Deutschen Patent- und Markenamt vorläufig das Symbol
G 06 F 3/14 der Internationalen Patentklassifikation erhalten.

München, den 25. November 1999

Deutsches Patent- und Markenamt

Der Präsident

Im Auftrag

Zeichen: 199 05 630.7

Dzierzon

Verfahren zur grafischen Darstellung und/oder Bearbeitung von Werten von Datentypen

Die Erfindung betrifft ein Verfahren zur grafischen Darstellung und/oder Bearbeitung von Werten von Datentypen einer formal definierten als Wertebaum vorliegenden Datenstruktur.

Programmier- und Spezifikationssprachen benötigen eine formale Syntax zur Beschreibung von Datentypen und deren Werten. Es sind daher Möglichkeiten bekannt, Datenstrukturen in einer formalen Notation unabhängig von einer bestimmten Sprache zu definieren. Die gemäß einer dieser Möglichkeiten definierten Datentypen können im allgemeinen beliebig komplex sein und sich zum Teil dynamisch ändern. Eine übersichtliche Darstellung solcher Datentypen zur Anzeige oder zur Eingabe von Werten durch den Benutzer ist daher notwendig. Dies ist insbesondere für das Management von Kommunikationsnetzen erforderlich, da dort häufig sehr komplexe Datenstrukturen bearbeitet werden.

In den meisten Sprachen und Syntaxnotationen bestehen bereits Regeln für die textuelle Darstellung von Werten. So wird beispielsweise bei ASN.1-Datentypen gemäß ITU-T Empfehlung X.208 häufig eine wenig übersichtliche Darstellung des Wertes als Zeichenfolge vorgenommen.

Bei einer bekannten Form der Darstellung als Baum ist die Struktur des Datentypes besser zu erkennen. Eine solche Darstellung ist in der Firmen-Druckschrift IBM: "IBM TMN Development Platform", Piscata NJ, USA, 1998 beschrieben. Ein anderes bekanntes Verfahren umfaßt die Zuordnung des Wertes eines Attributs zu einer grafischen Komponente durch manuelles Erstellen von Regeln, beispielsweise die Bestimmung der Farbe des grafischen Objektes durch den Wert des Datentypes, wie es beispielsweise in der Firmen-Druckschrift Objective Systems Integrators: "NetExpert Framework Overview", Folsom, CA, USA, 1997 beschrieben ist.

Da beliebige Definitionen von Datentypen möglich sind, kostet es bei den bekannten Verfahren viel Zeit, für jeden neuen Datentyp ein eigenes grafisches Benutzeroberflächen-Fenster zu programmieren.

Aufgabe der vorliegenden Erfindung ist es, diesen Zeitaufwand zu ersparen und trotzdem zu einer übersichtlichen Darstellung zu gelangen, die eine Überprüfung und gegebenenfalls eine Bearbeitung der Datenstruktur ermöglicht.

Diese Aufgabe wird mit dem erfindungsgemäßen Verfahren dadurch gelöst,

- daß der Datenstruktur ein Fenster als grafische Benutzeroberfläche zugeordnet wird,
- daß in dem Fenster generische skalierbare grafische Benutzeroberflächen-Komponenten hierarchisch eingefügt werden, wobei der Wertebaum der Datenstruktur auf die Benutzeroberflächen-Komponenten abgebildet wird,
- daß die grafischen Benutzeroberflächen-Komponenten in einer für den Benutzer erkennbaren Relation zu den Knoten des Wertebaums stehen und
- daß für jeden Teilbaum des Wertebaums eine grafische oder

...

textuelle Darstellung des Wertes wählbar ist.

Die übersichtliche Darstellung eines komplexen Datentyps mit dem erfindungsgemäßen Verfahren erspart dem Benutzer langes Suchen nach Definitionen und hilft, Fehler bei der Eingabe von Werten für diesen Datentyp zu vermeiden. Dabei bietet die Erfindung die Möglichkeit, redundante Informationen zu verbergen und wichtige Informationen ausführlich darzustellen, da bei dem erfindungsgemäßen Verfahren jeder Benutzer die Möglichkeit hat zu entscheiden, welche Informationen angezeigt werden sollen und für welche Informationen eine kompakte Darstellung ausreicht. Insbesondere erlaubt das erfindungsgemäße Verfahren bei der Bearbeitung der Datentypen eine einfache Wertzuweisung.

Eine besonders einfache und sichere Wertzuweisung kann gemäß einer Weiterbildung der Erfindung dadurch erfolgen, daß für eine Bearbeitung des Wertebaums für jeden Knoten eine Liste aller Werte vorgehalten wird, die zuweisungskompatibel mit dem dargestellten Datentyp sind, und daß für eine Wertzuweisung jeweils ein Wert aus der Liste selektiert wird. Dabei kann zur Vermeidung von Eingabefehlern vorgesehen sein, daß bei der Erstellung von Wertelisten die Zahl der in die Liste zu übernehmenden Werte gemäß zuvor definierter Regeln entsprechend des aktuellen Kontextes eingeschränkt wird.

Eine andere Weiterbildung der Erfindung besteht darin, daß bei einer Initialisierung der grafischen Benutzeroberfläche zunächst eine Visualisierung des Fensters vorgenommen wird und danach Daten initialisiert werden, die zu einer Bearbeitung vorgehalten werden, insbesondere Wertelisten. Durch den damit verbundenen schnelleren Bildaufbau kann der Benutzer bereits einen Überblick über die Datenstruktur gewinnen, bevor alle zur Darstellung und zur Bearbeitung des Datentyps erforderlichen Daten initialisiert sind.

...

Ein Datentyp läßt sich auch dann grafisch darstellen, wenn keine zusätzlichen, als Metadaten bezeichneten Informationen über den Datentyp in der grafischen Oberflächen-Applikation verfügbar sind, wenn gemäß einer anderen Weiterbildung der Erfindung der darzustellende Wert in einer Transfersyntax übergeben wird, die sämtliche für die Darstellung notwendigen Informationen bezüglich des Datentyps und der Wertbelegung enthält.

Eine vorteilhafte Ausführungsform des erfindungsgemäßen Verfahrens besteht darin, daß Datentypen, deren exakte Typenzuordnung gemäß dem Prinzip der späten Bindung erst zur Laufzeit ermittelt werden kann, als dynamisch änderbarer Teilbaum in dem durch die grafische Benutzeroberfläche dargestellten Wertebaum eingefügt werden. Dies hat den Vorteil, daß bei diesen Datentypen die Darstellung der aktuellen Wertbelegung nicht erst das Öffnen von Unterfenstern erfordert, sondern direkt im Hauptfenster erfolgen kann.

Eine andere vorteilhafte Ausführungsform sieht vor, daß für Datentypen, deren exakte Typenzuordnung gemäß dem Prinzip der späten Bindung erst zur Laufzeit durch die Markierung eines anderen Knotens bestimmt wird (beispielsweise "ANY DEFINED BY" in der Spezifikationssprache ASN.1), der Benutzer zur Eingabe aufgefordert wird, ob die Zuordnung automatisch oder nach einer manuellen Eingabe erfolgen soll. Damit ist auch eine Zuordnung möglich, wenn die zur automatischen Zuordnung benötigten Informationen nicht verfügbar sind.

Gemäß einer anderen vorteilhaften Weiterbildung ist vorgesehen, daß durch Zwischenspeicherung und Anklicken des jeweiligen Teilbaums Werte von einem Teilbaum in einen anderen Teilbaum übertragbar sind. Dabei ist zwar eine

...

Zuweisungs-Kompatibilität der den Teilbäumen zugeordneten Datentypen erforderlich. Die Datentypen müssen jedoch nicht innerhalb des gleichen Wertebaums instanziiert sein.

Eine einfache Benutzung zusammen mit anderen Programmen und das einfache Abbilden des allgemeinen Erscheinungsbildes ist gemäß einer anderen vorteilhaften Ausgestaltung dadurch möglich, daß das Verfahren durch ein oder mehrere in Anwenderprogramme einbindbare Programmodule realisiert wird.

Eine andere vorteilhafte Weiterbildung der Erfindung besteht darin, daß für jeden Knoten des Wertebaums, der durch den dargestellten Typ und die Relation zum übergeordneten Typ eindeutig benannt werden kann, zusätzliche anzuzeigende Informationen ablegbar sind. Dadurch können beispielsweise zusätzliche Textelemente für bestimmte Datentypen angezeigt werden, die zu einem beliebigen Zeitpunkt nach der Programmerstellung erzeugt werden und dynamisch zur Laufzeit in die Oberfläche eingebunden werden können.

Eine weitere Ausgestaltung der Erfindung besteht darin, daß während der Eingabe eines Wertes ständig geprüft wird, ob der eingegebene Wert für den zugehörigen Datentyp zulässig ist und ob der eingegebene Wert identisch mit dem derzeitigen aktuellen Wert des Datentyps ist, und daß das Ergebnis dem Benutzer kenntlich gemacht wird. Diese Ausgestaltung dient zu einer weiteren Erhöhung der Sicherheit und der Geschwindigkeit bei der Bearbeitung von Datentypen.

Ferner kann die Anzeige und/oder die Bearbeitung dadurch erleichtert werden, daß bereits während der Eingabe eines Wertes das Darstellungsformat veränderbar ist und somit beispielsweise ein Zahlenwert entweder dezimal oder binär dargestellt wird, bevor eine Wertübernahme in den Wertebaum erfolgt.

...

Ausführungsbeispiele der Erfindung sind in der Zeichnung anhand mehrerer Figuren dargestellt und in der nachfolgenden Beschreibung näher erläutert. Es zeigt:

Fig. 1 ein Fenster einer grafischen Benutzeroberfläche nach dem erfindungsgemäßen Verfahren,

Fig. 2 ein Beispiel der formalen Datendefinition gemäß ASN.1 und

Fig. 3 Prinzip einer generischen grafischen Darstellung.

Fig. 2 zeigt in Form eines strukturierten Textes die Definitionen der Datentypen "GetArgument" und "BaseManagedObjectClass". Der Typ "GetArgument" enthält zwei Komponenten "baseManagedObjectClass" und "baseManagedObjectInstance", wobei links die Namen und rechts die Typen der Komponenten, nämlich "BaseManagedObjectClass" und "GraphicString", angegeben sind. Der Typ "BaseManagedObjectClass" ist eine CHOICE mit ebenfalls zwei Komponenten, nämlich "globalForm" vom Typ "OBJECTIDENTIFIER" und "localForm" vom Typ "INTEGER".

Fig. 3 zeigt das Prinzip einer generischen grafischen Darstellung der in Fig. 2 dargestellten formalen Datendefinition. Dabei ist links der Wertebaum dargestellt, während rechts die grafische Repräsentation angedeutet ist. Dabei und im folgenden Text wird für grafische Benutzeroberfläche auch die Abkürzung GUI (= Graphical User Interface) verwendet. Zwischen den Blöcken verlaufende durchgezogene Linien bedeuten, daß die darunterliegenden Elemente in dem darüberliegenden Knoten enthalten sind, während gestrichelte Linien andeuten, daß jeweils die grafische Repräsentation in Relation zu dem jeweiligen Datentyp steht.

...

Fig. 1 zeigt die entsprechende grafische Benutzeroberfläche mit einem Fenster 1, in welchem alle enthaltenen Datentypen als grafische Benutzeroberflächen-Komponenten enthalten sind. Die Wurzel des Wertebaums (Fig. 3) ist mit GetArgument SEQUENCE angegeben. Die Eingabebalken 2, 3 und 4 mit den Etiketten "globalForm", "localForm" und "baseManagedObjectInstance" entsprechen den Blättern des Werte- bzw. GUI-Baumes. Mit Schaltflächen (Buttons) 5 bis 9 kann für jeden Knoten und jedes Blatt zwischen einer textuellen Darstellung (minus) und einer grafischen Darstellung (plus) gewählt werden.

Mit weiteren Schaltflächen 10, 11 (Radio Buttons) kann zwischen zwei alternativ zu bearbeitenden Datentypen (in diesem Fall Blätter) gewählt werden. Die Eingabebalken 2, 3, 4 sind jeweils mit einer Schaltfläche 12, 13, 14 versehen, mit welcher ein Fenster 15 zur Auswahl von als Konstanten abgelegten Werten geöffnet werden kann.

Die grafische Benutzeroberfläche weist ferner Schaltflächen 16, 17 zum Abschluß der Bearbeitung und Speicherung der bearbeiteten Daten sowie zum Verlassen der grafischen Benutzeroberfläche ohne Abspeicherung auf.

Ansprüche

1. Verfahren zur grafischen Darstellung von Werten von Datentypen einer formal definierten als Wertebaum vorliegenden Datenstruktur, dadurch gekennzeichnet,
 - daß der Datenstruktur ein Fenster als grafische Benutzeroberfläche zugeordnet wird,
 - daß in dem Fenster generische skalierbare grafische Benutzeroberflächen-Komponenten hierarchisch eingefügt werden, wobei der Wertebaum der Datenstruktur auf die Benutzeroberflächen-Komponenten abgebildet wird,
 - daß die grafischen Benutzeroberflächen-Komponenten in einer für den Benutzer erkennbaren Relation zu den Knoten des Wertebaums stehen und
 - daß für jeden Teilbaum des Wertebaums eine grafische oder textuelle Darstellung des Wertes wählbar ist.
2. Verfahren nach Anspruch 1, dadurch gekennzeichnet, daß für eine Bearbeitung des Wertebaums für jeden Knoten eine Liste aller Werte vorgehalten wird, die zuweisungskompatibel mit dem dargestellten Datentyp sind, und daß für eine Wertzuweisung jeweils ein Wert aus der Liste selektiert wird.
3. Verfahren nach Anspruch 2, dadurch gekennzeichnet, daß bei der Erstellung von Wertelisten die Zahl der in die Liste zu übernehmenden Werte gemäß zuvor definierter Regeln entsprechend des aktuellen Kontextes eingeschränkt wird.

...

4. Verfahren nach einem der vorhergehenden Ansprüche, dadurch gekennzeichnet, daß bei einer Initialisierung der grafischen Benutzeroberfläche zunächst eine Visualisierung des Fensters vorgenommen wird und danach Daten initialisiert werden, die zu einer Bearbeitung vorgehalten werden, insbesondere Wertelisten.

5. Verfahren nach einem der vorhergehenden Ansprüche, dadurch gekennzeichnet, daß der darzustellende Wert in einer Transfersyntax übergeben wird, die sämtliche für die Darstellung notwendigen Informationen bezüglich des Datentyps und der Wertbelegung enthält.

6. Verfahren nach einem der vorhergehenden Ansprüche, dadurch gekennzeichnet, daß Datentypen, deren exakte Typenzuordnung gemäß dem Prinzip der späten Bindung erst zur Laufzeit ermittelt werden kann, als dynamisch änderbarer Teilbaum in dem durch die grafische Benutzeroberfläche dargestellten Wertebaum eingefügt werden.

7. Verfahren nach einem der vorhergehenden Ansprüche, dadurch gekennzeichnet, daß für Datentypen, deren exakte Typenzuordnung gemäß dem Prinzip der späten Bindung erst zur Laufzeit durch die Markierung eines anderen Knotens bestimmt wird (beispielsweise "ANY DEFINED BY" in der Spezifikationssprache ASN.1), der Benutzer zur Eingabe aufgefordert wird, ob die Zuordnung automatisch oder nach einer manuellen Eingabe erfolgen soll.

8. Verfahren nach einem der vorhergehenden Ansprüche, dadurch gekennzeichnet, daß durch Zwischenspeicherung und Anklicken des jeweiligen Teilbaums Werte von einem Teilbaum in einen anderen Teilbaum übertragbar sind.

...

9. Verfahren nach einem der vorhergehenden Ansprüche, dadurch gekennzeichnet, daß das Verfahren durch ein oder mehrere in Anwenderprogramme einbindbare Programmodule realisiert wird.

10. Verfahren nach einem der vorhergehenden Ansprüche, dadurch gekennzeichnet, daß für jeden Knoten des Wertebaums, der durch den dargestellten Typ und die Relation zum übergeordneten Typ eindeutig benannt werden kann, zusätzliche anzeigende Informationen ablegbar sind.

11. Verfahren nach einem der vorhergehenden Ansprüche, dadurch gekennzeichnet, daß während der Eingabe eines Wertes ständig geprüft wird, ob der eingegebene Wert für den zugehörigen Datentyp zulässig ist und ob der eingegebene Wert identisch mit dem derzeitig aktuellen Wert des Datentyps ist, und daß das Ergebnis dem Benutzer kenntlich gemacht wird.

12. Verfahren nach einem der vorhergehenden Ansprüche, dadurch gekennzeichnet, daß bereits während der Eingabe eines Wertes das Darstellungsformat veränderbar ist und somit beispielsweise ein Zahlenwert entweder dezimal oder binär dargestellt wird, bevor eine Wertübernahme in den Wertebaum erfolgt.

Zusammenfassung

Bei einem Verfahren zur grafischen Darstellung und/oder Bearbeitung von Werten von Datentypen einer formal definierten als Wertebaum vorliegenden Datenstruktur wird der Datenstruktur ein Fenster als grafische Benutzeroberfläche zugeordnet. In dem Fenster werden generische skalierbare grafische Benutzeroberflächen-Komponenten hierarchisch eingefügt, wobei der Wertebaum der Datenstruktur auf die Benutzeroberflächen-Komponenten abgebildet wird. Die grafischen Benutzeroberflächen-Komponenten stehen in einer für den Benutzer erkennbaren Relation zu den Knoten des Wertebaums. Für jeden Teilbaum des Wertebaums ist eine grafische oder textuelle Darstellung des Wertes wählbar.

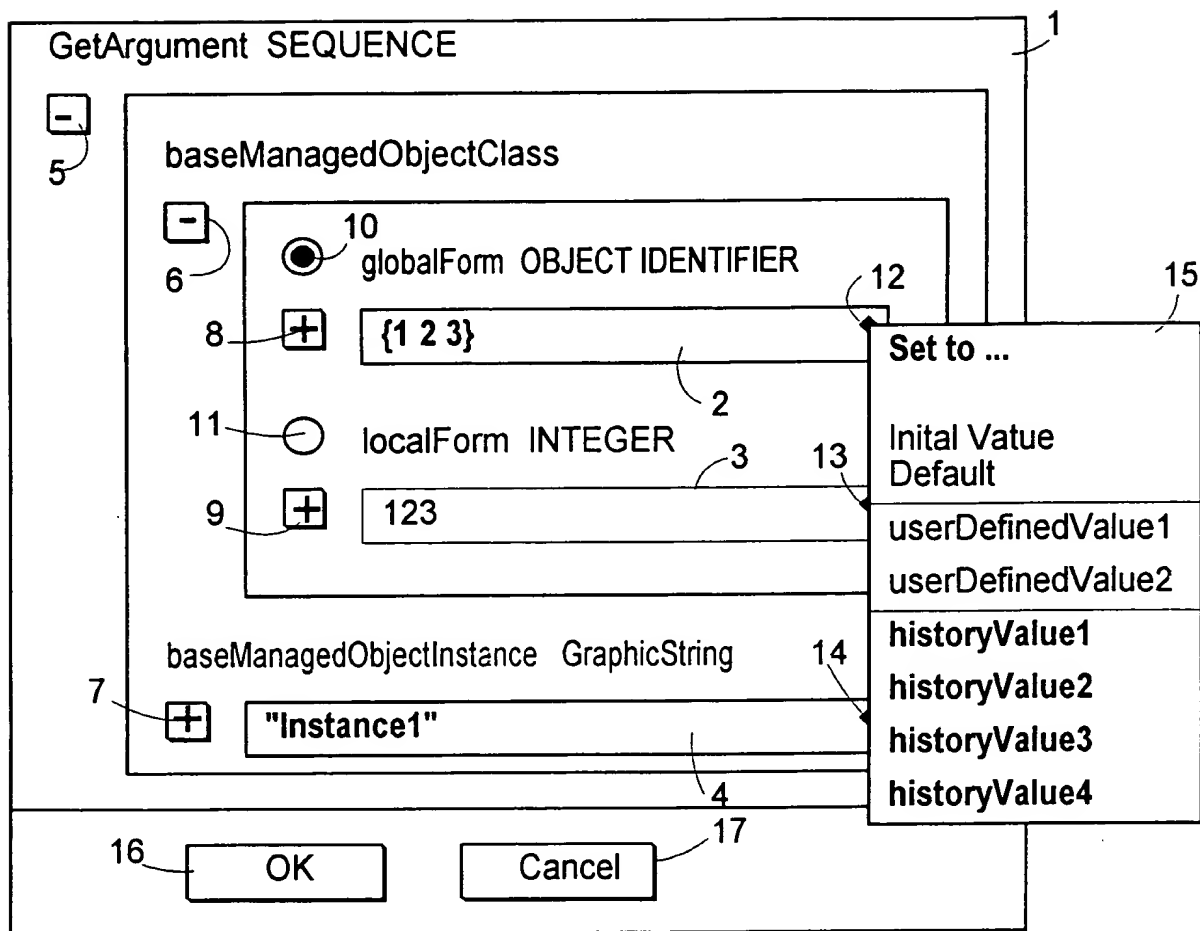


Fig.1

```

GetArgument ::= SEQUENCE {
    baseManagedObjectClass      BaseManagedObjectClass
    baseManagedObjectInstance  GraphicString
}
BaseManagedObjectClass ::= CHOICE {
    globalForm      OBJECT IDENTIFIER
    localForm       INTEGER
}

```

Fig.2

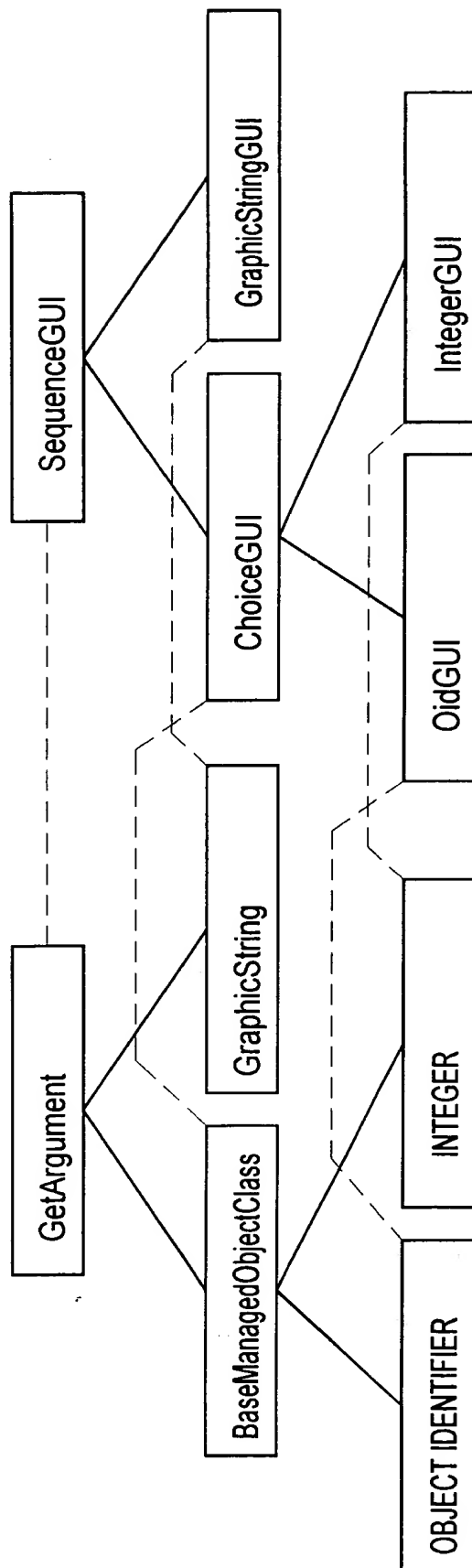


Fig.3